Как правильно убить MySQL?



Света Смирнова
Percona

Владимир Федорков,
ECOMMPAY

HighLoad++, Москва, Оффлайн, 2021





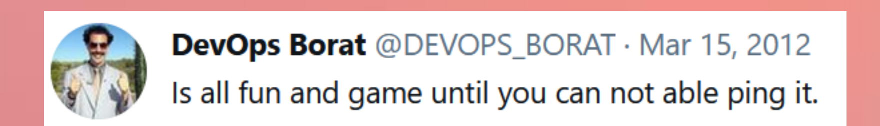
Генеральный партнер





DISCLAMER / НЕ ПОВТОРЯЙТЕ ЭТИ СОВЕТЫ!

Примеры приведены в сугубо образовательных целях и никогда не должны рассматриваться как руководство к действию. Их применение может привести (и приводило ранее) к катастрофическим последствиям.

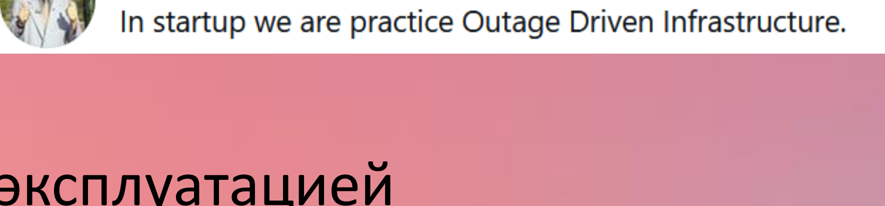






Почему этот доклад здесь?!

- Изначально должны были говорить
  - про обеспечение стабильности
  - про прозрачность системы
  - про коммуникацию между разработкой и эксплуатацией
- Но!
  - Очень много документации
  - Постоянные доклады на конференциях
    - На этой тоже есть.
  - Никому не интересно.
- Было принято тяжелое решение сменить таргет аудиторию
- Будут только боевые, проверенные примеры





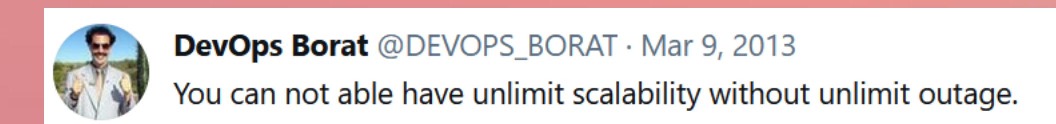


А вы точно знаете как правильно?

- Света Смирнова
- Автор книги MySQL Troubleshooting
- Работа в поддержке компаний
  - MySQL AB, Sun, Oracle
  - Сейчас в Percona
- Карьеру начинала как веб разработчик
  - Работала с ОРМ
  - За что и расплачивается

- Владимир Федорков
- Первый MySQL 3.23.32 поставил в 2001ом году
  - Пытался изменить с Oracle и PostgeSQL
    - Не зашло
- Со второй половины 200х работаю с MySQL и высокими нагрузками
- С октября 2019-го занимаюсь MySQL в ECOMMPAY
  - За это время совместными усилиями сделали прод стабильным
    - Подробности были вчера в главном зале

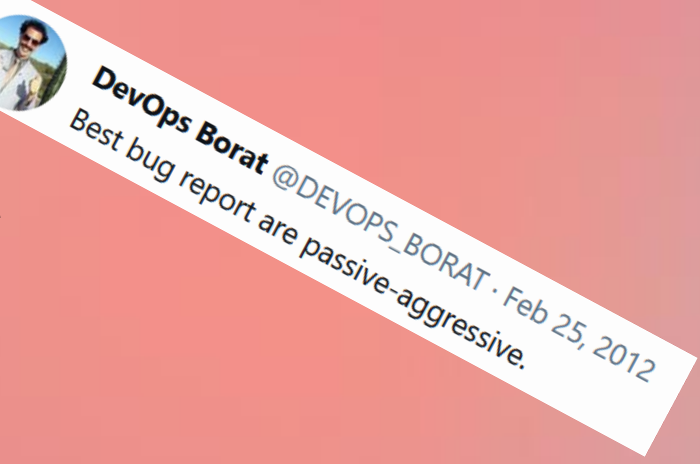






Убить MySQL сложно.

- Система отказоустойчивая, быстрая и довольно надежная.
- Простые запросы и простая логика отрабатывают быстро
  - 95%th времени выполнения запросов в нашем случае 400 наносекунд
- Для того, что бы работало медленно и неустойчиво нужно последовательно усложнять
  - Конфигурацию
  - Запросы
  - Архитектуру приложения в целом







DevOps Borat @DEVOPS_BORAT · Jan 6, 2012
In devops record of wiki still usable is 43 day.

Конфигурация

- Дает возможность MySQL (да и любой базе) реализовать свой потенциал на текущем железе.
- Наша задача – сразу не дать MySQL никакого шанса
- Путей два
  - Недоконфигурить
  - Переконфигурить





Вариант I: Вообще без конфигурации!

- Неплохой вариант
  - Innodb_buffer_pool = 128Mb,
- По началу все будет работать
- Потом тоже, но не очень.
- Плюсы: не нужно ничего делать
- Минусы: легко понять, что конфига нет

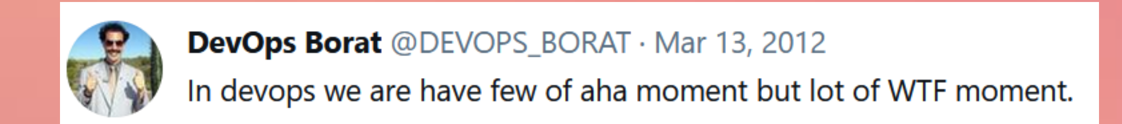






Вариант II: Как можно больше конфигурации

- Меняем все значения по умолчанию, не ленимся
- Минимум 350 параметров, что бы запутать противника
- Цель: ни одного значения по умолчанию!
- Важно помнить: наши помыслы чисты, все для результата!



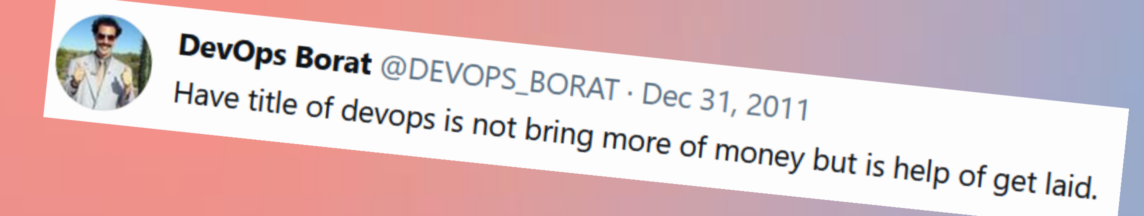
- Три места в которые мы можем упереться
  - Память
  - Диск
  - Процессор





Отвратительная настройка памяти

- Память быстрая – все буфферы как можно больше
- OOM наш друг и соратник
- На данные не смотрим
- Значения по умолчанию не для нас
  - Важно потом объяснить зачем меняли







Диск – безопасно и быстро





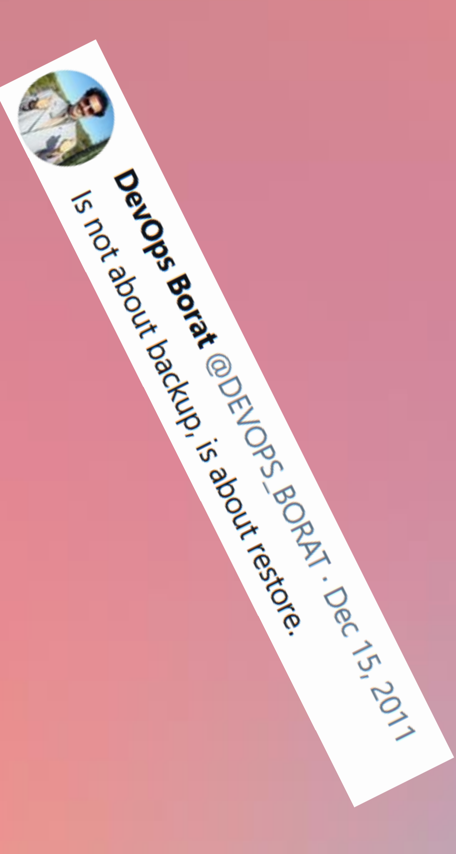
Диск – ~~безопасно~~ и ~~быстро~~





Диск – опасно и медленно!

- MySQL пишет данные на диск 4 раза
  - Везде включаем fsync
    - Но только где не нужно
- Тротлим запись и чтение для innodb
  - innodb_io_read_threads
  - innodb_io_write_threads
- Ломаем транзакционную систему
  - innodb_log_file_size в минимум
- Бинлоги
  - max_binlog_size, sync_binlog







Остается процессор!

- Тротлим использование потоков
  - innodb_thread_concurrency в минимум
- Включаем query cache
- Основной удар по процессору могут дать только запросы
  - Особенно если данные пока помещаются в память

DevOps Borat @DEVOPS_BORAT · Feb 11, 2012
I am only lose hope when developer is tell me "I get it done soon".

ecommpay



Запросы

- Обеспечивают 99,99% тормозов
  - И головную боль админов
- Лучший MySQL запрос тот который до MySQL не дошел.
- MySQL не сможет выполнить некоторые запросы быстро
  - Просто потому что так спроектирован
- Как написать тормозной запрос?







Обманываем оптимизатор!

- Заворачиваем все колонки в функции
- SELECT * FROM table WHERE DAY(FROM_UNIXTIME(`ts`)) = 205
- … WHERE DATE_ADD(td, INTERVAL 1 HOUR) > NOW()
- … WHERE (c_int + 20) = 21
- … WHERE HEX(CAST(vbin1 ((-1 & c_int) << 5) % 0)) = 0x0;
- **DevOps Borat** @DEVOPS_BORAT · Jan 11, 2013 ···
  Software project 1) On time 2) On budget 3) With quality. You can not able pick any.





Ограничение дизайна СУБД

- ACID медленный!
- B-Tree работает только для спец. случаев
- Запрос обрабатывается одним ядром
- Есть Full-text индексы, они медленные
- Про репликацию можно говорить часами
- Как написать тормозной запрос ?





Не даем никакого шанса на индексы

- Никогда не используем равенство!
- SELECT * FROM table WHERE deleted != 1
- SELECT * FROM table WHERE id NOT IN (1,2,3,...,10)
- SELECT * FROM table WHERE id IN (… МНОГО!...)
- Помните, B-Tree это префиксный индекс!
- WHERE url LIKE ‘%чтототам%’
  - Не путать с LIKE ‘чтототам%’!
- Даже если индекс есть, стараемся делать range scan



DevOps Borat @DEVOPS_BORAT · Feb 16, 2012 ···

In devops we are follow 80/20 rule: we are automate 80% and we are leave hard 20% manual for job security.





Почему full table scan это дорого?

- Читать всю таблицу долго
- Пока читаем – вымываем память базы ненужными данными
- Грузим диск
- Грузим процессор
- Греем атмосферу, потребляем электричество!
  - Запросы крутятся – лавеха мутится!
- Заставляют использовать индексы?

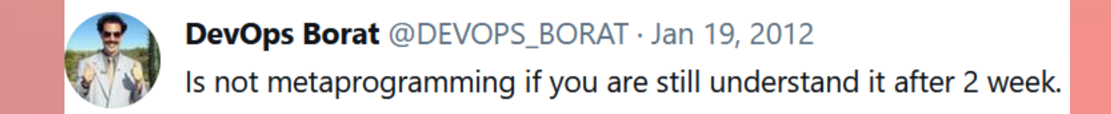






Высокоселективные запросы

- SELECT * / COUNT(*) FROM users WHERE active=1
  - Здравствуй, половина таблицы!
- Индексы не помогают
- Другие варианты
  - Агрегировать SUM(), AVG() на клиенте

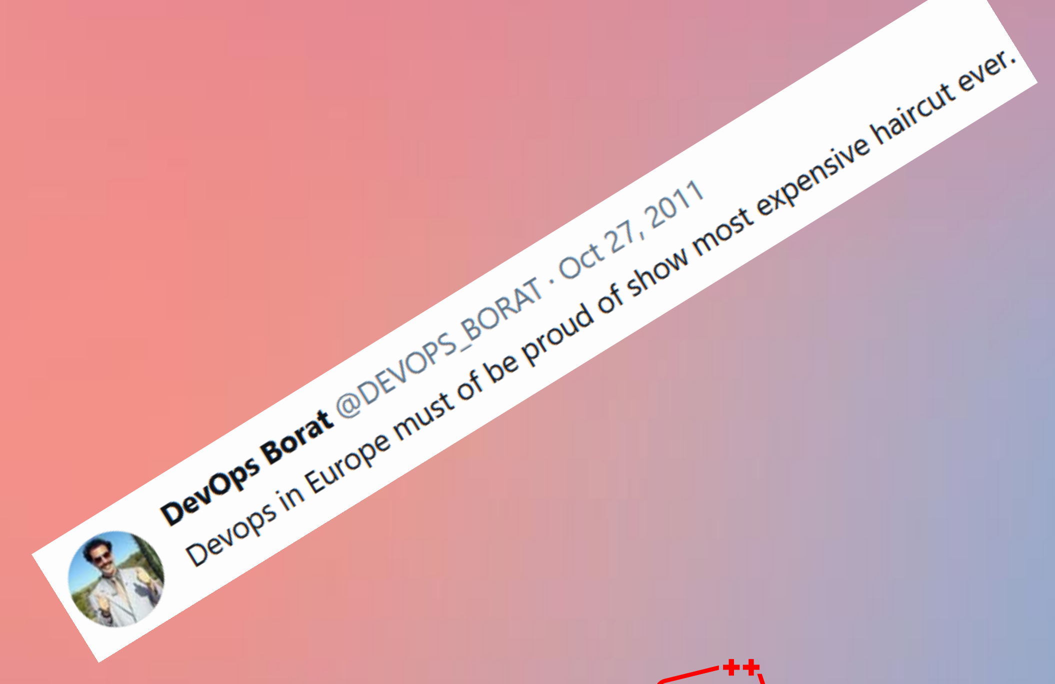






LIMIT 1 OFFSET 100000000000000000000

- Заставляет вычитывать всю таблицу даже если индексы есть
- Выкидывает все данные кроме последней страницы
- Вымывает все данные из бафферпула
- Полностью эмулирует full table scan

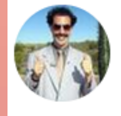






У нас есть победитель!

ORDER BY RAND()



DevOps Borat @DEVOPS_BORAT · Jan 26, 2012
In devops we are spend 1 hour on Twitter for each of 1 sec of make test suite faster.

- Даже если стоит SELECT * ORDER BY RAND() LIMIT 1;

*также подойдет любая сортировка не по индексу где есть большая выборка





Писать плохие запросы руками опасно



- В гите все видно
  - Лучше конечно отказаться от версионирования.
- Могут найти и заставить переделать
  - С одной стороны это дает шанс сделать запрос еще хуже!
  - Но ведь могут отревьювить!
- Выход есть!





Используем ORM!

- Готовые запросы любой ужасности!
- Вы не виноваты!
- Исправить невозможно!
- У вас же модель данных!!!







Больше логики приложения в базе!

- Foreign keys / Cascading
- Триггеры тоже неплохо.
- Хранимые процедуры – сразу победа!



DevOps Borat @DEVOPS_BORAT · Dec 17, 2011 ···
Big Data Analytic is show 60% of email is spam and 20% can ignore. Other 20% is Nagios alert.





Репликация!

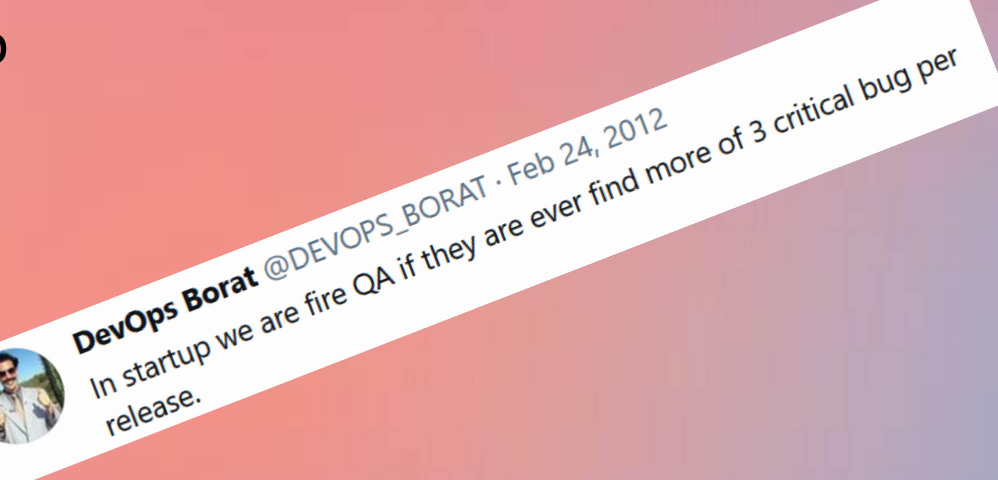
- Кольцевая репликация
  - Желательно без GTID
- Большие пачки на запись
- Пригодятся любые тяжелые миграции
- **DevOps Borat** @DEVOPS_BORAT · Jan 3, 2013
  1+1=2 only in legacy system. In modern distributed database with eventual consistent is 1+1=1.





Подводя итоги

- Конфигурация быстрое – но ненадежное решение
  - Можно найти и быстро поменять
- Запросы – надежнее
  - Сложнее исправить
  - Дольше переписать
  - Поменять логику иногда очень дорого
    - Особенно, если у вас ORM

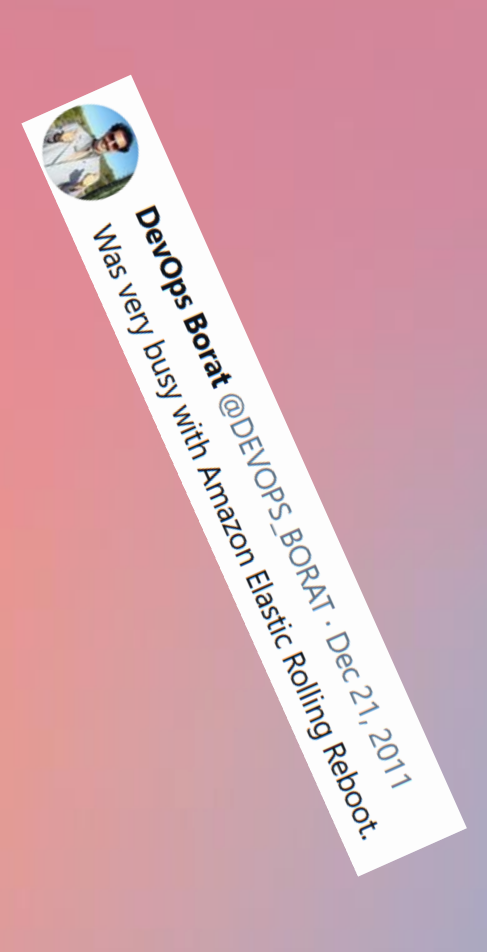






Важно вовремя не увидеть проблему

- EXPLAIN для лузеров
- Мониторинг лишний
- Графики не в коем случае!
- Про PMM забыть сразу: сложно, трудно и дорого!
  - Графики вообще непонятные!
  - [можно без этого]







Другие мудрости

- Увидел – промолчи!
    - Подари товарищу незабываемую ночь!
- Лучший деплой – в пятницу вечером!
- Самые надежные миграции – в пиковое время
- Работы лучше делать когда дежурные уже ушли спать
- Проснулся ночью – поработай!







Почему то, о чем мы сегодня говорим – важно?

- Ковид увеличил нарузку на online системы
- Электронные компоненты становятся дефицитными
- Сроки поставок сдвигаются
- Стоимость идет резко вверх, а деньги дешевеют
- Стоимость масштабирования растёт
- Необходимость оптимизации будет вставать все острее



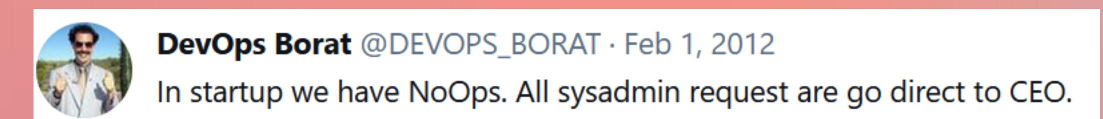
DevOps Borat @DEVOPS_BORAT · Apr 19, 2012
5% of devops are have infrastructure as code. Rest of 95% are have infrastructure as PowerPoint.





Что делать?

- Конференции
  - Вы по адресу!
- Книги
  - MySQL Troubleshooting
  - MySQL по максимуму (High Performance MySQL)
- Online вебинары и доклады
- Практика-практика-практика.
- И конечно почитать https://twitter.com/devops_borat

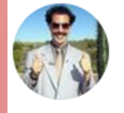






Где получить практику?

- Работая с реальным проектом
- В грамотной команде
- Первое время с наставником
  - Который будет тебя тащить
- Решать сложные для себя задачи

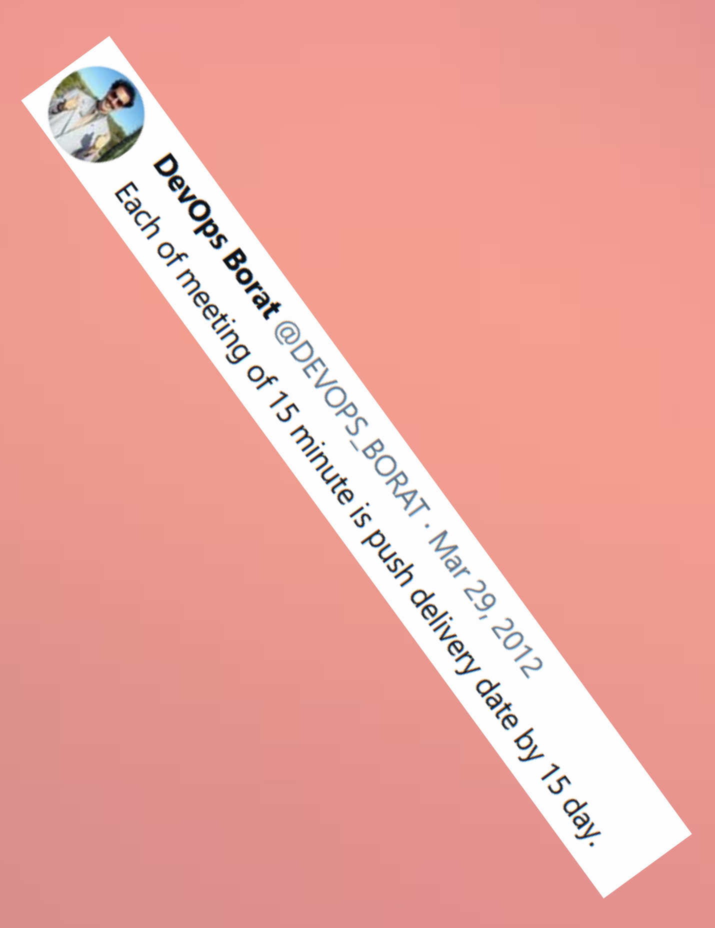


DevOps Borat @DEVOPS_BORAT · Mar 17, 2012

If you are not wake up of Nagios minimum 2 time per week, your technology is not sufficient cutting-edge.






DevOps Borat @DEVOPS_BORAT · Mar 29, 2012
Each of meeting of 15 minute is push delivery date by 15 day.

Вопросы!


ecommpay



DevOps Borat @DEVOPS_BORAT · Mar 29, 2012 ···

In devops we have success is 10% inspiration and 90% Not Invented Here.

Спасибо!

